Untitled

2024-12-01

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(glmnet)
```

```
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
##
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
##
## Loaded glmnet 4.1-8
```

```
library(survival)
library(ggplot2)
library(splines)
library(reshape2)
```

```
##
## Attaching package: 'reshape2'
##
## The following object is masked from 'package:tidyr':
##
##     smiths
```

```
library(nlme)
```

```
##
## Attaching package: 'nlme'
```

```
##
## The following object is masked from 'package:dplyr':
##
##     collapse
```

```
library(survminer)
```

```
## Loading required package: ggpubr
##
## Attaching package: 'survminer'
##
## The following object is masked from 'package:survival':
##
##     myeloma
```

```
library(geepack)
library(ggbiplot)
```

```
## Loading required package: plyr
## ------------------------------------------------------------------------------
## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
## ------------------------------------------------------------------------------
##
## Attaching package: 'plyr'
##
## The following object is masked from 'package:ggpubr':
##
##     mutate
##
## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
##
## The following object is masked from 'package:purrr':
##
##     compact
##
## Loading required package: scales
##
## Attaching package: 'scales'
##
## The following object is masked from 'package:purrr':
##
##     discard
##
## The following object is masked from 'package:readr':
##
##     col_factor
##
## Loading required package: grid
```

```r
library(caret)
```

```
## Loading required package: lattice
##
## Attaching package: 'caret'
##
## The following object is masked from 'package:survival':
##
##     cluster
##
## The following object is masked from 'package:purrr':
##
##     lift
```

```r
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
##
## Attaching package: 'pROC'
##
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```r
library(plotROC)
```

```
##
## Attaching package: 'plotROC'
##
## The following object is masked from 'package:pROC':
##
##     ggroc
```

```r
library(survivalROC)
library(timeROC)
```

```r
METABRIC_RNA_Mutation <- read_csv("dat_cleaned.csv")
```

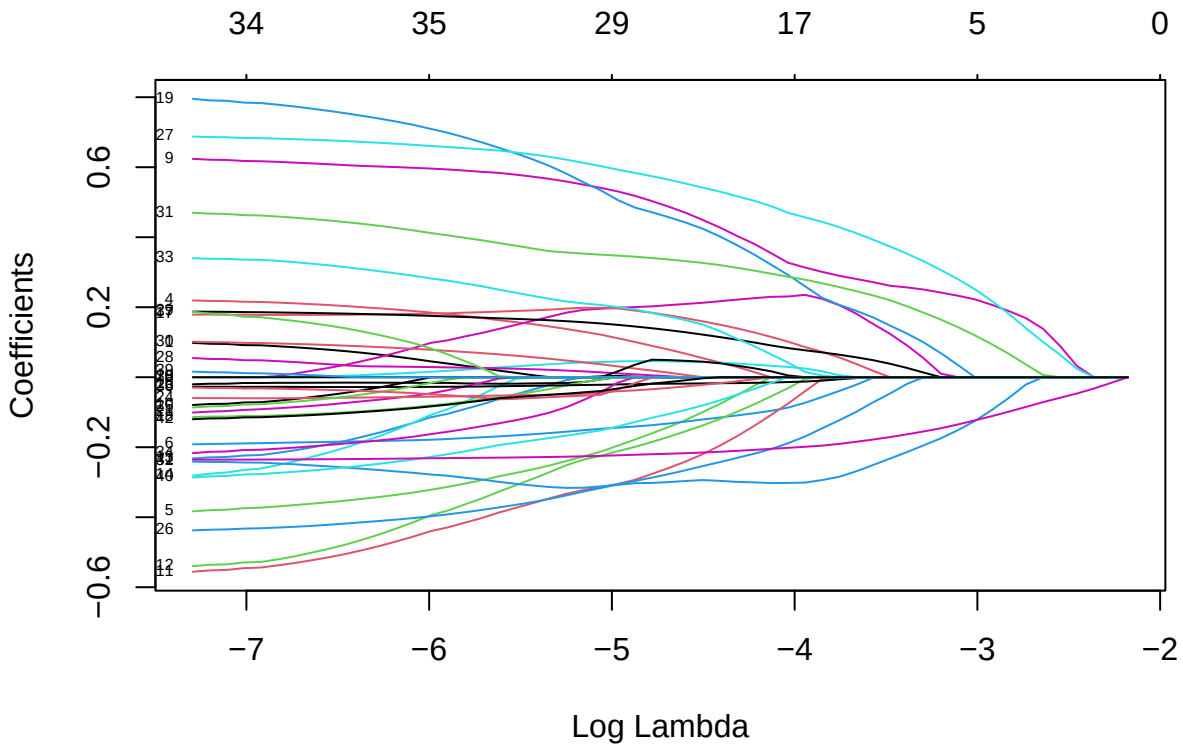
```
## Warning: One or more parsing issues, call `problems()` on your data frame for details,
## e.g.:
##   dat <- vroom(...)
##   problems(dat)

## Rows: 1904 Columns: 693
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (186): type_of_breast_surgery, cancer_type, cancer_type_detailed, cellul...
## dbl (507): patient_id, age_at_diagnosis, chemotherapy, cohort, neoplasm_hist...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
# colnames(METABRIC_RNA_Mutation)

# clinical factors
clinical <-  METABRIC_RNA_Mutation[,1:31]

clinical[] <- lapply(clinical, function(x) if(is.character(x)) as.factor(x) else x)

clinical <- clinical |>
  mutate(chemotherapy = as.factor(chemotherapy),
         cohort = as.factor(cohort),
         neoplasm_histologic_grade = as.factor(neoplasm_histologic_grade),
         hormone_therapy = as.factor(hormone_therapy),
         radio_therapy = as.factor(radio_therapy),
         tumor_stage = as.factor(tumor_stage)) |>
  select(-death_from_cancer, -cohort, -er_status_measured_by_ihc, -her2_status_measured_by_snp6, -oncoti
  na.omit()

clinical_factor <- clinical |>
  select(-patient_id, -overall_survival_months, -overall_survival)

colnames(clinical_factor)
```

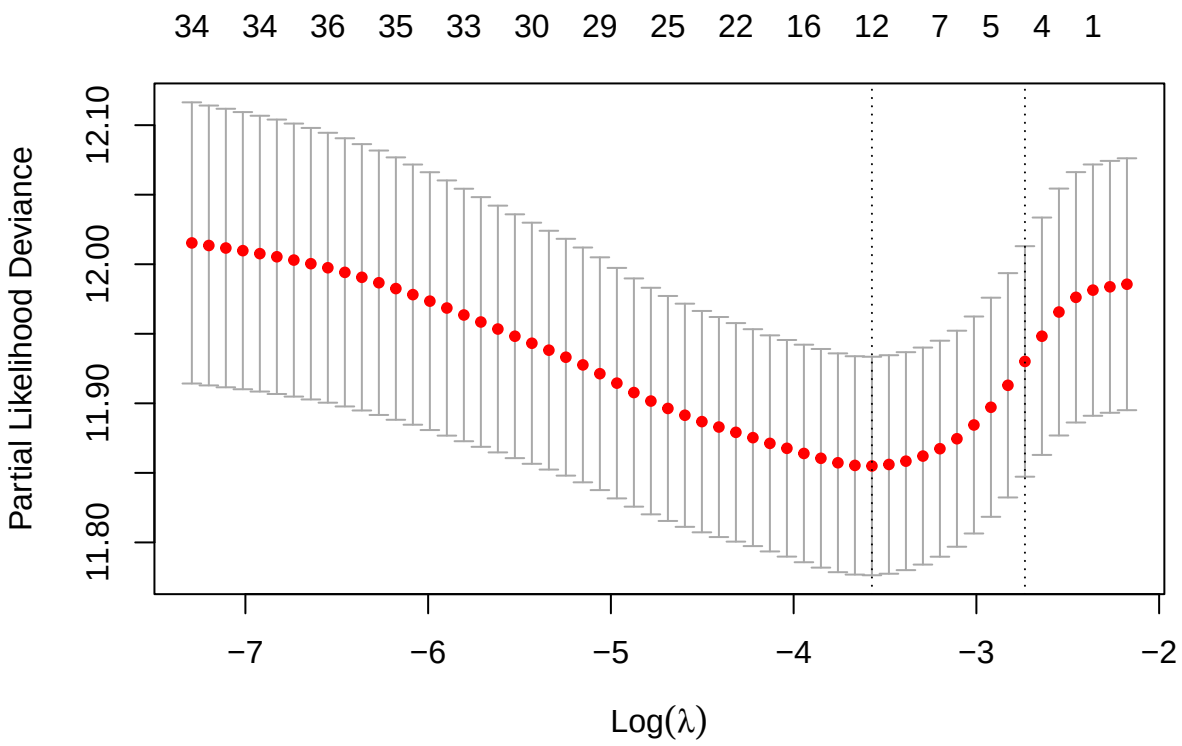
```
##  [1] "age_at_diagnosis"              "type_of_breast_surgery"
##  [3] "cancer_type"                   "cancer_type_detailed"
##  [5] "cellularity"                   "chemotherapy"
##  [7] "pam50_._claudin.low_subtype"   "er_status"
##  [9] "neoplasm_histologic_grade"     "her2_status"
## [11] "tumor_other_histologic_subtype" "hormone_therapy"
## [13] "inferred_menopausal_state"     "primary_tumor_laterality"
## [15] "lymph_nodes_examined_positive" "mutation_count"
## [17] "pr_status"                     "radio_therapy"
## [19] "X3.gene_classifier_subtype"    "tumor_size"
## [21] "tumor_stage"
```

```r
clinical_data_matrix <- model.matrix(~., data = clinical_factor |> select(-age_at_diagnosis,
                                                                          -lymph_nodes_examined_positive
                                                                          -mutation_count,
                                                                          -tumor_size))[, -1]

clinical_data_matrix_cont <- as.data.frame(scale(clinical_factor|> select(age_at_diagnosis,
                                                                          lymph_nodes_examined_positive
                                                                          mutation_count,
                                                                          tumor_size)))

clinical_df <- as.matrix(cbind(clinical_data_matrix, clinical_data_matrix_cont))

# Survival time and event
time <- clinical$overall_survival_months
status <- clinical$overall_survival
y <- Surv(time = time, event = status)

# Lasso Cox with no lambda =0, without regularization = coxph_fit
fit_clinical = glmnet(clinical_df,y,
```

```
              alpha = 1,family = "cox")
plot(fit_clinical,xvar= 'lambda',label = TRUE)
```




```
# CV for 10-fold
set.seed(2024)
lasso_clinical <- cv.glmnet(clinical_df, y, family = "cox", alpha = 1, nfolds = 10)

plot(lasso_clinical)
```




```
# selected clinical factors
best_lambda_clinical <- lasso_clinical$lambda.min

# final lasso fit
final_clinical_model <- glmnet(clinical_df, y, family = "cox", alpha = 1, lambda = best_lambda_clinical)
coefficient <- coef(lasso_clinical, best_lambda_clinical)
selected_index <- which(as.numeric(coefficient) != 0)
selected_features <- names(coefficient[selected_index,])
```

(1. did not included the variable "death_from_cancer", as I think it is a little bit hard to interpreat 2. did not included the variable "cohort", as I think this just a group ID for patients 3. did not included the variable "er_status_measured_by_ihc", as I think this may inidcate similar clinical factor as "er_status" did, so I only keep "er_status". 4. did not included the variable "her2_status_measured_by_snp6", as I think this may inidcate similar clinical factor as "her2_status" did and I only keep "her2_status" 5. did not included "oncotree_code", as it is a code for standardizing cancer type diagnosis from a clinical perspective assigned by another source: Memorial Sloan Kettering Cancer Center (MSK) 6. did not included the variable "integrative_cluster", as it described the cancer type based on some gene expression, I think this may be redundant with selected genes. In addition, variable "3-gene_classifier_subtype" and tumor_other_histologic_subtype may also decribed cancer subtypes based on Three Gene classifier and microscopic examination.) 7. did not included the variable "nottingham_prognostic_index" as it is highly correlated with the survival time.

The selection for clinical factor is based on 686 patients with complete data. Then for clinical variables, we choose a total of 11 clinical factors that may have potential effects on the survival prognosis, namely `cancer_type_detailed`, `chemotherapy`, `pam50_+_claudin-low_subtype`, `neoplasm_histologic_grade`, `her2_status`, `tumor_other_histologic_subtype`, `hormone_therapy`, `primary_tumor_laterality`,

radio_therapy, X3.gene_classifier_subtype and mutation_count

for DEG genes

m-RNA levels z-score for 331 genes (with subfamily because total 489 genes)

not including mutation genes

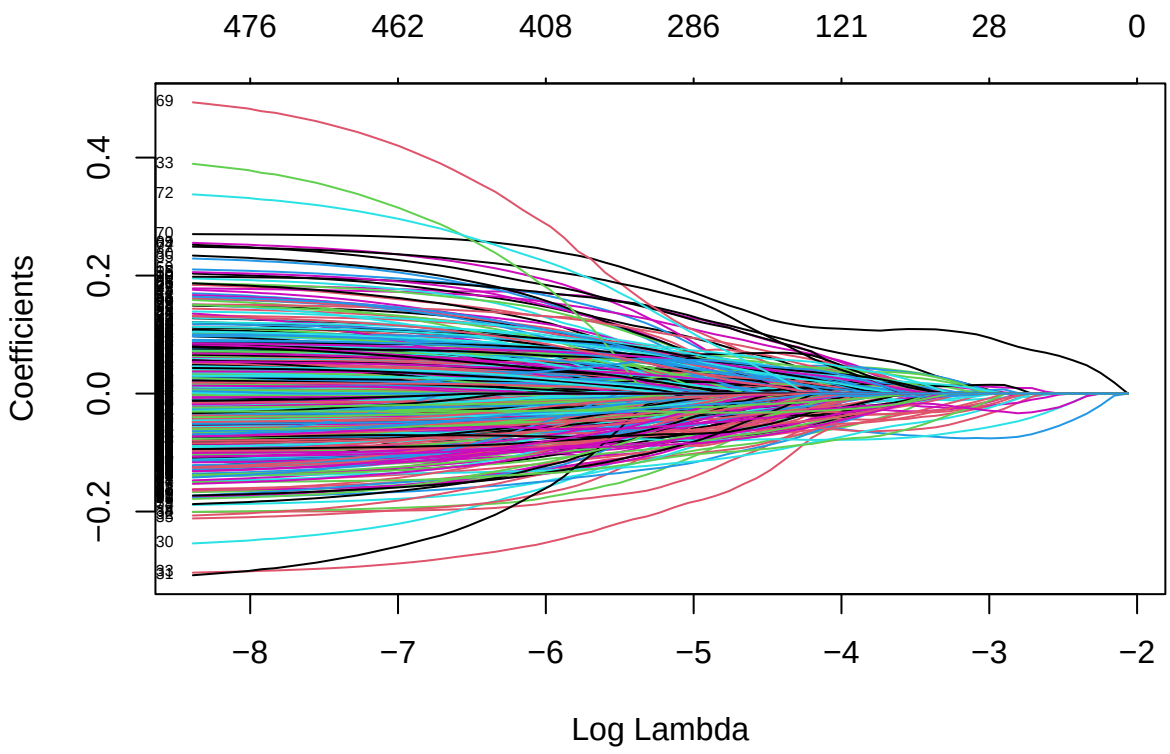
the survival event in this data 1 = living and 0 = dead

```
# Survival time and event
METABRIC_RNA_Mutation <- read_csv("dat_cleaned.csv")
```

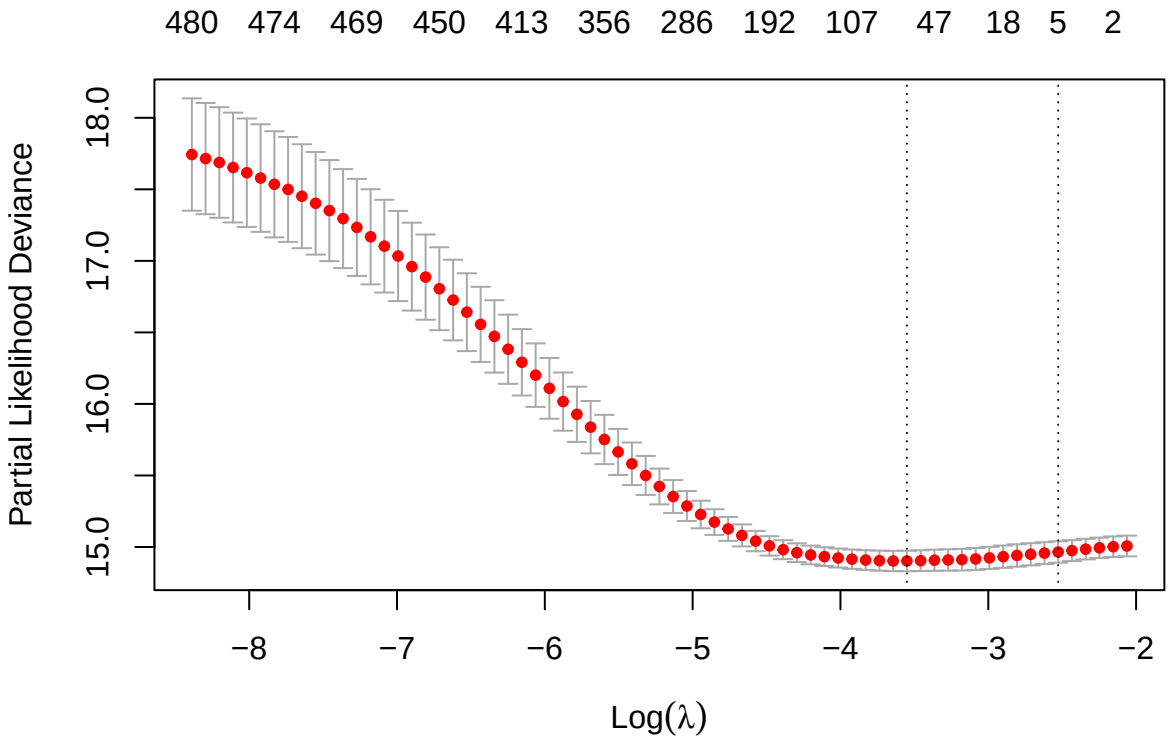
```
## Warning: One or more parsing issues, call `problems()` on your data frame for details,
## e.g.:
##   dat <- vroom(...)
##   problems(dat)
```

```
## Rows: 1904 Columns: 693
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (186): type_of_breast_surgery, cancer_type, cancer_type_detailed, cellul...
## dbl (507): patient_id, age_at_diagnosis, chemotherapy, cohort, neoplasm_hist...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
METABRIC_RNA_Mutation = METABRIC_RNA_Mutation[which(METABRIC_RNA_Mutation$overall_survival_months != 0)
time <- METABRIC_RNA_Mutation$overall_survival_months
METABRIC_RNA_Mutation$overall_survival <-  ifelse(METABRIC_RNA_Mutation$overall_survival == 1, 0, 1)
status <- METABRIC_RNA_Mutation$overall_survival
y <- Surv(time = time, event = status)

# for genes based on all patients with positive survival (omit 171)
gene_expression_data <- METABRIC_RNA_Mutation[32:520]  |> na.omit()
gene_expression_data <- as.matrix(gene_expression_data)

# Lasso Cox with no lambda =0, without regularization = coxph_fit
fit = glmnet(gene_expression_data,y,
             alpha = 1,family = "cox")
plot(fit,xvar= 'lambda',label = TRUE)
```




```r
# CV for 5-fold
set.seed(2024)
lasso_gene <- cv.glmnet(gene_expression_data, y, family = "cox", alpha = 1, nfolds = 5)

plot(lasso_gene)
```




```
# selected clinical factors
best_lambda_gene <- lasso_gene$lambda.1se

# final lasso fit
final_gene_model <- glmnet(gene_expression_data, y, family = "cox", alpha = 1, lambda = best_lambda_gene
coefficient <- coef(lasso_gene, best_lambda_gene)
selected_index <- which(as.numeric(coefficient) != 0)
selected_genes <- names(coefficient[selected_index,])

# save results
write_rds(selected_genes, "selected_5_genes.rds")
write_rds(selected_features, "selected_11_clinical.rds")
```

Finally selected 5 genes by 1SE method: "selected_genes" (including genes which belonged in the same family)

risk model

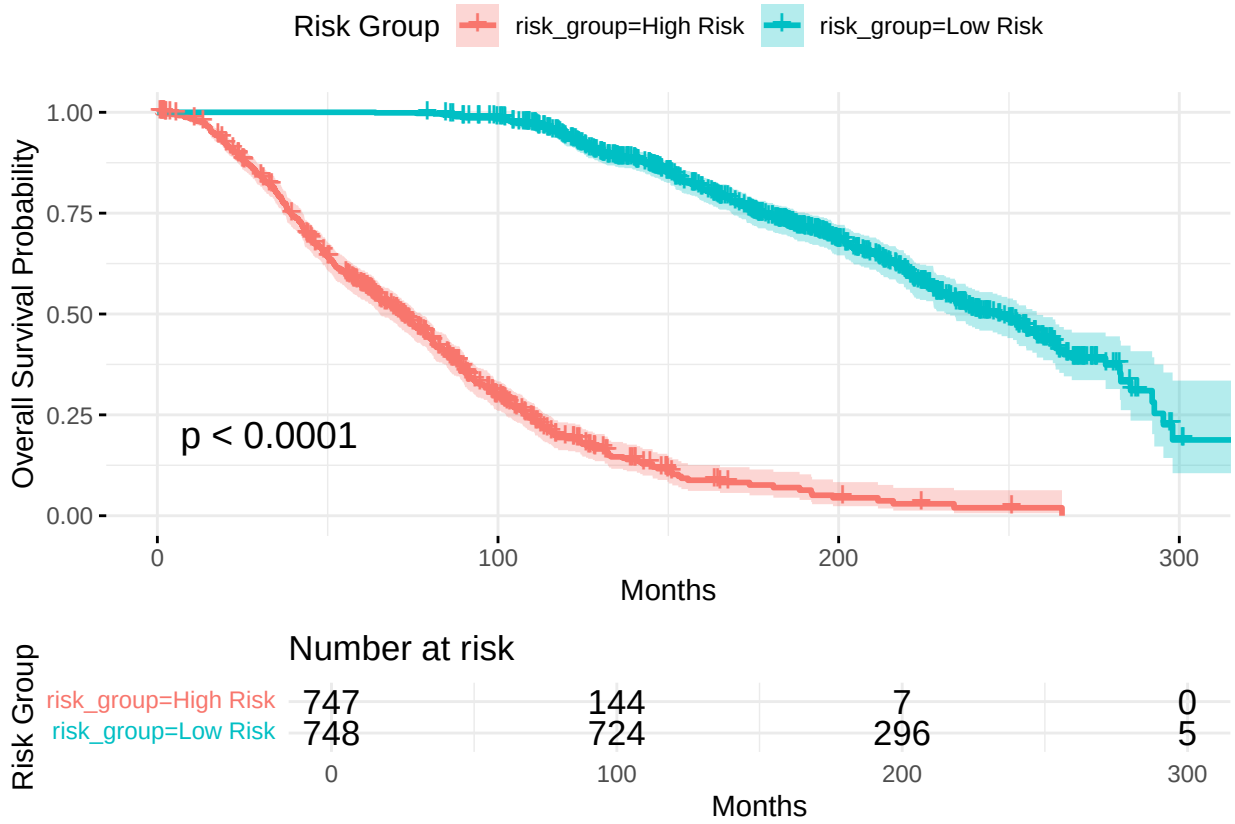
```
# Load data
METABRIC_RNA_Mutation = read_csv("dat_cleaned.csv")
```

```
## Warning: One or more parsing issues, call `problems()` on your data frame for details,
## e.g.:
```

```
##   dat <- vroom(...)
##   problems(dat)

## Rows: 1904 Columns: 693
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (186): type_of_breast_surgery, cancer_type, cancer_type_detailed, cellul...
## dbl (507): patient_id, age_at_diagnosis, chemotherapy, cohort, neoplasm_hist...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
# Load selected covariates, adjust as necessary if these change!
selected_clinical_vars = c(
  "cancer_type_detailed",
  "chemotherapy",
  "pam50_._claudin.low_subtype",
  "neoplasm_histologic_grade",
  "her2_status",
  "tumor_other_histologic_subtype",
  "hormone_therapy",
  "primary_tumor_laterality",
  "radio_therapy",
  "X3.gene_classifier_subtype",
  "mutation_count"
)

selected_gene_vars = read_rds("selected_5_genes.rds")


# Clinical variables
clinical_data = METABRIC_RNA_Mutation |>
  select(
    patient_id,
    overall_survival_months,
    overall_survival,
    all_of(selected_clinical_vars)
  )

# Gene expression variables
gene_data = METABRIC_RNA_Mutation |>
  select(patient_id, all_of(selected_gene_vars))

# Merge clinical and gene data
final_data = clinical_data |>
  inner_join(gene_data, by = "patient_id") |>
  na.omit()

categorical_vars = c(
  "cancer_type_detailed",
  "chemotherapy",
  "pam50_._claudin.low_subtype",
  "her2_status",
  "tumor_other_histologic_subtype",
```

```
  "hormone_therapy",
  "primary_tumor_laterality",
  "radio_therapy",
  "X3.gene_classifier_subtype"
)

continuous_vars = c(
  "mutation_count",
  selected_gene_vars
)

final_data[categorical_vars] = lapply(final_data[categorical_vars], as.factor)

final_data[continuous_vars] = lapply(final_data[continuous_vars], as.numeric)

final_data$overall_survival = ifelse(final_data$overall_survival == 1, 0, 1)

# risk_score model
final_cox_model <- readRDS("~/Documents/2024 class/survive/final_project/final_cox_model.rds")

risk_scores = predict(final_cox_model, newdata = final_data, type = "risk")
final_data$risk_scores = risk_scores

final_data$risk_group <- ifelse(final_data$risk_scores > median(final_data$risk_scores), "High Risk", "

# Kaplan-Meier
surv_obj <- Surv(time = final_data$overall_survival_months, event = final_data$overall_survival)
km_fit <- survfit(surv_obj ~ risk_group, data = final_data)

ggsurvplot(
  km_fit,
  data = final_data,
  pval = TRUE,
  conf.int = TRUE,
  risk.table = TRUE,
  legend.title = "Risk Group",
  xlab = "Months",
  ylab = "Overall Survival Probability",
  ggtheme = theme_minimal()
)
```

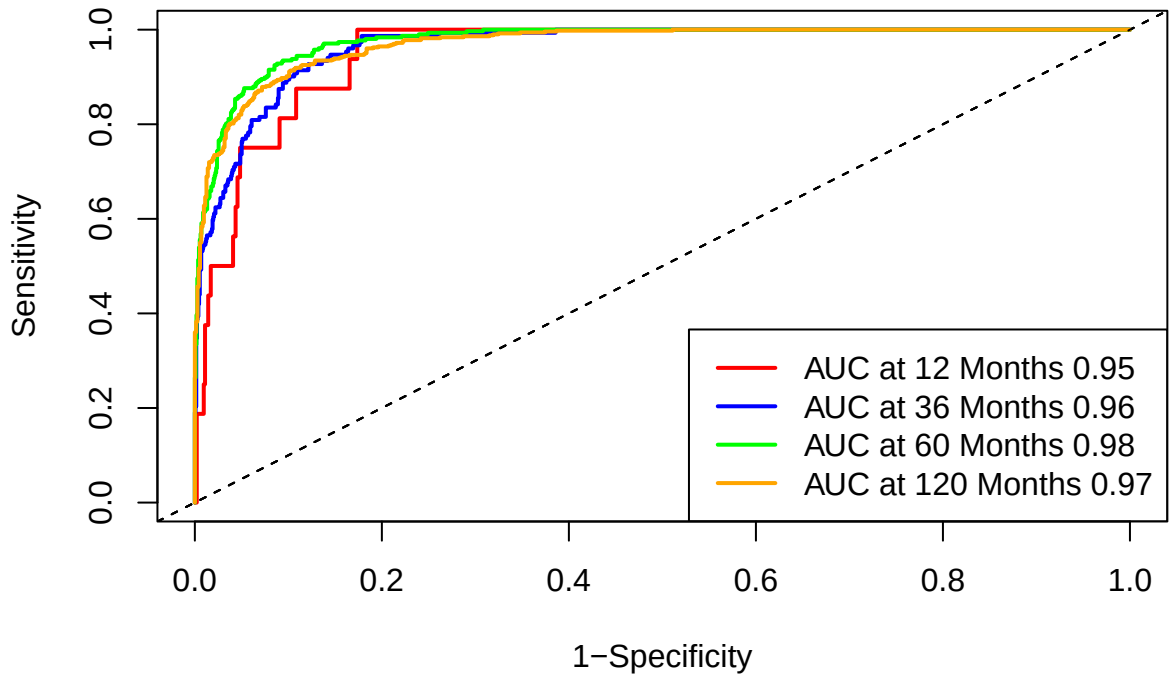



```r
# performance
# Time-Dependent ROC Curves
time_points <- c(12, 36, 60, 120)
roc_time <- timeROC(
  T = final_data$overall_survival_months,
  delta = final_data$overall_survival,
  marker = final_data$risk_scores,
  cause = 1,
  times = time_points,
  iid = TRUE
)

# AUC
roc_time$AUC
```

```
##      t=12      t=36      t=60     t=120
## 0.9509710 0.9644161 0.9759414 0.9697232
```

```r
#plot
colors <- c("red", "blue", "green", "orange")

# 1 year
plot(
  roc_time,
  time = time_points[1],
```

```
  col = colors[1],
  title = "Time-Dependent ROC Curves",
  lwd = 2
)

# add others
for (i in 2:length(time_points)) {
  plot(
    roc_time,
    time = time_points[i],
    col = colors[i],
    add = TRUE,
    lwd = 2
  )
}

# legend
legend(
  "bottomright",
  legend = paste("AUC at", time_points, "Months", round(roc_time$AUC,2)),
  col = colors,
  lty = 1,
  lwd = 2,
)
```



summary

variable selection:

To avoid constructed a too complex model that may lead to over fitting, we first performed a variable selection process to filter both clinical and genomic data.

For selecting clinical variables: Based on the complete data, to select clinical variables that have potential effect on the survival prognostic of breast cancer patients, we performed LASSO Cox regression, which may control the complexity of model by penalty parameter λ. Among 28 clinical features. After 10 fold cross validation, we choose the λ which reached the minimal partial likelihood deviance and finally selected 11 of them and included them in our model building part.

For selecting genome data: The selection process for genome data is based on the around 500 gene expression profile of a total of 1903 patients with positive overall survival month. As a result of LASSO Cox selection step with 5 fold cross validation, a total of 5 genes were selected by the optimal λ using 1SE method. (The simplest model within the allowable error range (usually 1 standard error) corresponds to a more sparse feature selection.)

Outcome Analysis

risk stratification After building the final Cox Proportional Hazards model, as mentioned before, if can be used to predict the risk of patients with breast cancer, then we applied this Cox model as a risk model and calculated the risk scores for each patients based on their own clinical data and gene expression profiles and further clustered patients into high risk and low risk group and from the survival analysis, we found significant difference in survival outcomes between these two risk groups, which may suggesting that patients with different respond to clinical treatment, heterogeneous clinical features, as well as differential expressed genes may have impacted on their survival prognosis and can be captured by our cox Proportional Hazards model. In addition, this risk model performance is evaluated by time depended ROC and AUC. Again from the result, we can see that our model performed well in patients' survival prognosis, especially for predicting the risk of patients.

Overall outcome In conclusion, after variable selection and constructed the Cox Proportional Hazards model based on clinical and gnome data of breast cancer patients, on the one hand, we found that several clinical factors such as the mutation count of related genes in patients, tumor sub types, as well as some treatments may have significant impact on patients survival condition. In addition, the interaction between aggressive level of tumor cells and time also indicate significant influence on patients' prognosis. which may suggesting that the contribution of neoplasm histologic grade to survival risk showed a downward trend overtime. In other words, this may indicate that the effect of tumor histological grade is gradually diminished during long-term follow-up.

On the other hand, three out of 5 selected genes show significance based on the model estimated results, including Signal transducer and activator of transcription-5A (STAT5A), glycogen synthase kinase 3β (gsk3b) and ATP Binding Cassette Subfamily B Member 1 (abcb1), which found to be related with acceleration of breast cancer cells motility, tumor invasion and proliferation in recent study, and may also responsible for acquired chemotherapy resistance [ref*].

Furthermore, our cox model may also acted as a useful tool to predicted patients' risk scores and may have protential clinical applications on decision-making.

ref*: https://breast-cancer-research.biomedcentral.com/articles/10.1186/s13058-019-1125-0 https://pmc.ncbi.nlm.nih.gov/articles/PMC10487083/ https://www.nature.com/articles/s41375-018-0117-x